What is claimed is:

1. A method for calculating a local mean number of tasks for each processing element (PE_r) in a parallel processing system, wherein each processing element (PE_r) has a local number of tasks associated therewith and wherein r represents the number for a selected processing element, the method comprising:

assigning a value (E_r) to said each processing element (PE_r);

summing a total number of tasks present on said parallel processing system and said value (E_r) for said each processing element (PE_r);

dividing the sum of said total number of tasks present on said parallel processing system and said value (E_r) for said each processing element (PE_r) by a total number of processing elements in said parallel processing system; and

truncating a fractional portion of said divided sum for said each processing element.

2. The method of claim 1 wherein said assigning a value (E_r) to said each processing element (PE_r) comprises setting said value (E_r) equal to a number between 0 and ($N-1$), where N represents said total number of processing elements in said parallel processing system.

3. The method of claim 2 wherein said assigning a value (E_r) to said each processing element (PE_r) further comprises giving a unique number to said each value (E_r) for said each processing element PE_r.

4. The method of claim 1 wherein said assigning a value (E_r) to said each processing element (PE_r) comprises setting said value (E_r) equal to said number for a selected processing element (r).

5. The method of claim 1 wherein said value (E_r) controls said truncating step such that said total number of tasks for said parallel processing system equals the sum of said local mean number of tasks for each processing element (PE_r) in said parallel processing system.

6. The method of claim 1 wherein said local mean number of tasks for each processing element (PE_r) within said parallel processing system is equal to one of X and ($X+1$).

7. The method of claim 1 wherein said assigning step, said summing step, said dividing step, and said truncating step are completed on a portion of said parallel processing system.

8. The method of claim 1 wherein said assigning step, said summing step, said dividing step, and said truncating step are completed on a line of said processing elements within said parallel processing system.

9. The method of claim 1 wherein said assigning step, said summing step, said dividing step, and said truncating step are completed on a loop of said processing elements within said parallel processing system.

10. The method of claim 1 wherein said assigning step, said summing step, said dividing step, and said truncating step are completed on an array of said processing elements within said parallel processing system.

11. The method of claim 1 wherein said assigning step, said summing step, said dividing step, and said truncating step are completed on an array of two or more interconnected processing elements within said parallel processing system.

12. A memory device carrying a set of instructions which, when executed, perform a method comprising:

assigning a value (E_r) to said each processing element (PE_r);

summing a total number of tasks present on said parallel processing system and said value (E_r) for said each processing element (PE_r);

dividing the sum of said total number of tasks present on said parallel processing system and said value (E_r) for said each processing element (PE_r) by a total number of processing elements in said parallel processing system; and

truncating a fractional portion of said divided sum for said each processing element.